

processing logic. The presence, or not, of question controls **308A** and answer controls **310A** determines whether speech output or recognition input is enabled on activation. Command controls **310B** and user initiative answers are activated by specification of the Scope property on the answer controls **310A** and command controls **310B**.

[0075] In simple voice-only applications, a QA control **320** will typically hold one question control or object **308A** and one answer control or object **310A**. Although not shown in the example below, command controls **310B** may also be specified, e.g. Help, Repeat, Cancel, etc., to enable user input which does not directly relate to the answering of a particular question.

[0076] A typical 'regular' QA control for voice-only dialog is as follows:

---

```
<Speech:QA
  id="QA_WhichOne"
  ControlsToSpeechEnable="textBox1"
  runat="server" >
  <Question >
    <prompt> Which one do you want?
  </prompt >
  </Question>
  <Answer >
    <grammar src="whichOne.gram" />
  </Answer>
</Speech:QA>
```

---

[0077] (The examples provided herein are written in the ASP.Net framework by example only and should not be considered as limiting the present invention.)

[0078] In this example, the QA control can be identified by its "id", while the association of the QA control with the desired primary or visual control is obtained through the parameter "ControlsToSpeechEnable", which identifies one or more primary controls by their respective identifiers. If desired, other well-known techniques can be used to form the association. For instance, direct, implicit associations are available through the first and second approaches described above, or separate tables can be created used to maintain the associations. The parameter "runat" instructs the web server that this code should be executed at the webserver **202** to generate the correct markup.

[0079] A QA control might also hold only a statement control **308C**, in which case it is a prompt-only control without active grammars (e.g. for a welcome prompt). Similarly a QA control might hold only an answer control **310A**, in which case it may be a multimodal control, whose answer control **310A** activates its grammars directly as the result of an event from the GUI, or a scoped mechanism (discussed below) for user initiative.

[0080] It should also be noted that a QA control **320** may also hold multiple output controls **308** and input controls **310** such as multiple question controls **308A** and multiple answers controls **310A**. This allows an author to describe interactional flow about the same entity within the same QA control. This is particularly useful for more complex voice-only dialogs. So a mini-dialog which may involve different kinds of question and answer (e.g. asking, confirming, giving help, etc.), can be specified within the wrapper of the

QA control associated with the visual control which represents the dialog entity. A complex QA control is illustrated in FIG. 11.

[0081] The foregoing represent the main features of the QA control. Each feature is described from a functional perspective below.

[0082] Answer Control

[0083] The answer control **310A** abstracts the notion of grammars, binding and other recognition processing into a single object or control. Answer controls **310A** can be used to specify a set of possible grammars relevant to a question, along with binding declarations and relevant scripts. Answer controls for multimodal applications such as "Tap-and-Talk" are activated and deactivated by GUI browser events. The following example illustrates an answer control **310A** used in a multimodal application to select a departure city on the "mouseDown" event of the textbox "txtDepCity", and write its value into the primary textbox control:

---

```
<Speech:QA
  controlsToSpeechEnable="txtDepCity"
  runat="server" >
  <Answer id="AnsDepCity"
    StartEvent="onMouseDown"
    StopEvent="onMouseUp"
  />
  <grammar src="/grammars/depCities.gram"/>
  <bind value="//sml/DepCity"
    targetElement="txtCity" />
  </Answer>
</Speech:QA>
```

---

[0084] Typical answer controls **310A** in voice-only applications are activated directly by question controls **308A** as described below.

[0085] The answer control further includes a mechanism to associate a received result with the primary controls. Herein, binding places the values in the primary controls; however, in another embodiment the association mechanism may allow the primary control to look at or otherwise access the recognized results.

[0086] Question Control

[0087] Question controls **308A** abstracts the notion of the prompt tags (Appendix A) into an object which contains a selection of possible prompts and the answer controls **310A** which are considered responses to the question. Each question control **308A** is able to specify which answer control **310A** it activates on its execution. This permits appropriate response grammars to be bundled into answer controls **310A**, which reflect relevant question controls **308A**.

[0088] The following question control **308A** might be used in a voice-only application to ask for a Departure City:

---

```
<Speech:QA id="QADepCity"
  controlsToSpeechEnable="txtDepCity"
  runat="server" >
  <Question id="Q1" Answers="AnsDepCity" >
    <prompt>
      Please give me the departure
```

---